Video Popularity Dynamics and Its Implication for Replication

Yipeng Zhou, *Member, IEEE*, Liang Chen, Chunfeng Yang, and Dah Ming Chiu, *Fellow, IEEE*

***Abstract*—Popular online video-on-demand (VoD) services all maintain a large catalog of videos for their users to access. The knowledge of video popularity is very important for system operation, such as video caching on content distribution network (CDN) servers. The video popularity distribution at a given time is quite well understood. We study how the video popularity changes with time, for different types of videos, and apply the results to design video caching strategies. Our study is based on analyzing the video access levels over time, based on data provided by a large video service provider. Our main finding is, while there are variations, the glory days of a video's popularity typically pass by quickly and the probability of replaying a video by the same user is low. The reason appears to be due to fairly regular number of users and view time per day for each user, and continuous arrival of new videos. All these facts will affect how video popularity changes, hence also affect the optimal video caching strategy. Based on the observation from our measurement study, we propose a mixed replication strategy (of LFU and FIFO) that can handle different kinds of videos. Offline strategy assuming tomorrow's video popularity is known in advance is used as a performance benchmark. Through trace-driven simulation, we show that the caching performance achieved by the mixed strategy is very close to the performance achieved by the offline strategy.**

***Index Terms*—Dynamic video popularity, lifetime, video cache, video replication.**

I. Introduction

THE popular online VoD services, such as YouTube,[1] Youku,[2] Tencent Video,[3] and Netflix,[4] attract a lot of users. They typically use content distribution networks (CDNs), with many edge servers spread out in areas close to the users. An important operational decision is to assign the servers to serve a huge and growing catalog of videos by caching right videos on edge servers. This decision process can benefit greatly from a good understanding of video popularity and how it changes over time. In our study, video popularity can be measured by the relative number of views of a video.

Most of past works focused on the study of video popularity by taking a static view [1]. Static popularity is defined by the relative number of views of each video over a given time period. To ensure the computed static popularity is sufficiently accurate and reliable, the measurement period is usually sufficiently long. Most often, static popularity distribution is found to follow the Zipf Law [2]. For live streaming channels and TV channels, it is most natural to use the static model of popularity since the channel content is updated daily, and their popularity can be compared on a daily basis.

However, for videos in a real-world VoD system, the situation is complicated. The upload of a video in the VoD system occurs at different times. And people usually lose their interests in viewed videos because their time is taken by newly updated content. Thus, the relative online video popularity should be dynamic and keep changing with time, depending on how many days the video has already been in the system, and the rate of new videos that compete for viewership are added to the system. This motivates us to study the dynamics of a video's popularity as it goes through its life cycle, starting as a newly arrived video gaining popularity, and then entering a waning phase, and then losing popularity to newer videos. The knowledge of the dynamics of video popularity can have a lot of applications, such as optimizing bandwidth resource allocation, designing strategies for video replication on edge servers and users' local storage, and recommending videos. If the evolution of video popularity is predictable in advance, we can replicate more copies of the videos with increasing popularity and reduce replication of videos losing popularity to make space for the hot videos. It can also be used to create a more realistic workload model than the static queueing network model in [3] used for studying resource allocation strategies. Video popularity is naturally involved in video recommendation [4]. Based on the knowledge of popularity changes, we can design smarter recommendation algorithms and caching algorithms. However, the study of the changes of video popularity is a very challenging task, since the catalog of videos in the system at any one time is very large, and the computation of popularity requires us to consider all videos together, and we must do the computation over a sufficiently long period of time.

Actually, how users select videos, i.e. user behavior, is the core issue to determine video popularity. However, user behavior depends on various exogenous parameters that are difficult to be sketched clearly. For example, all users' view behavior is restricted by their limited available time that can be

Manuscript received March 17, 2015; accepted June 02, 2015. Date of publication June 18, 2015; date of current version July 15, 2015. This work was supported in part by the Foundation of Shenzhen City under Grant KQCX20140519103756206, in part by the Hong Kong RGC under Grant 14201814, and in part by the Natural Science Foundation of China under Grant 61402297. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Yiannis Andreopoulos. *(Corresponding author: Liang Chen.)*

Y. Zhou is with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518000, China (e-mail: ypzhou@szu.edu.cn).

L. Chen is with the College of Information Engineering, Shenzhen University, Shenzhen 518000, China (e-mail: lchen@szu.edu.cn).

C. Yang and D. M. Chiu are with the Department of Information Engineering, Chinese University of Hong Kong, Hong Kong 999077 (e-mail: yc012@ie.cuhk.edu.hk; dmchiu@ie.cuhk.edu.hk).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMM.2015.2447277

[1][Online]. Available: http://www.youtube.com

[2][Online]. Available: http://www.youku.com/

[3][Online]. Available: http://v.qq.com/

[4][Online]. Available: http://www.netflix.com/

spent on viewing videos. This is regarded as *constrained eyeball* for brevity in the following discussion. In addition to constrained eyeball, user behavior could also be affected by the number of videos relative to the number of users, the rate new videos arrive and so on. In this work, we study the dynamics of video popularity from three dimensions: popularity's age-sensitivity, constrained eyeball and probability of replay. Our study is based on the viewing records collected by one of the largest online VoD providers in China, Tencent Video . Due to the extremely large quantity of data, both in number of users and number of videos, we can only process records within a month. Since the lifetime of most videos are relatively short, this analysis in a limited time span is already quite revealing.

On the basis of the observations summarized from the measurement study, we design a video caching strategy on edge servers by combining Least-Frequently Used (LFU) and First-in-First-out (FIFO) strategies. We use the trace log collected from Tencent Video as well to evaluate the caching strategy. By comparing the hit rate with the offline strategy which knows future video popularity in advance, we find that the performance of the mixed strategy is very close to that of the offline strategy. Note, the concept for dynamic popularity we are developing is not contradict with static popularity which can be understood as the popularity computed over a sufficiently long period of time. But the latter is not practical since online video services usually need to make close-to-realtime decisions for resource (including bandwidth and storage) allocation and recommendation, dynamic video popularity is used instead to derive information necessary for decision making.

The rest of the paper is organized as follows. In Section II, we review how data is collected. Then, the measurement study is given in Section III and Section IV. Video caching strategies and the trace-driven simulation results are presented in Section V. Before we conclude the paper, related works are discussed in Section VI.

II. Data Collection and Classification

Tencent Video, our collaborator, is one of the largest online VoD service providers in China. Macroscopically, Tencent Video supports more than 50 millon active users each day. During busy hours, there are more than 2 million concurrent users. At the same time, Tencent Video provides a large catalog of videos of different types, including movie, TV episodes, music video (MV), entertainment videos, as well as short clips of news and sports. We select five most representative video types: *News*, *Sports*, *TV*, *MV* and *Movies* for our study. These five types take more than one half of total daily views. The number of each type of videos are on the order of 100,000. We choose to treat these videos based on their types since dynamics of their popularity are expected to be different. For example, News should be time sensitive since few people are interested in old news. We believe the above five types are quite representative - all other types are similar to one of the above five. For example, based on our measurement, TV series and Animes have almost the same popularity distribution on either daily basis or weekly basis.[5] We have excluded user generated content (UGC) as one of the types we study, since there exist several prior works on that type of videos already [5], [6]. For the above five types, there are already hundreds of millions of viewing sessions in this platform on a daily basis.

TABLE I
Proportion of Views for Five Types of Videos in the Sampled Dataset

Types	Movie	TV	MV	Sport	News
Prop. (%)	5.94	23.09	18.05	6.32	46.6

Here, we briefly describe how the data is collected. The Web video players (Flash and HTML5) used in Tencent Video service are programmed to submit view reports to a cloud server. The view report contains when and which a video are accessed by which user, and how long the view session lasts. This is done on a continuous basis, and billions of view records (the whole data size is around 750 GB in our collection) were available for us to analyze. Besides user view reports, there is a separate media database that keeps track of the new videos added to the service each day and what type each video belongs to. Like other *Big Data* problems, the challenge is how to process the vast amount of data to derive useful results.

III. Video Popularity in a Time Window

As we discussed in the introduction, one characterization of view popularity can be defined as the number of users who viewed the video. This, however, is not easy to compute for the large number of videos at hand, since it would theoretically require to keep huge number of view records for a very long time. In practice, video popularity is measured in terms of view count (number of views a video received) within a time window. The time window can be a day, a week, or longer. We get more precise video popularity by using longer time window. However, given the time and computing resources available, we used a week as the time window.

First, let us look at the proportion of views taken by each type of videos at the aggregate level. The percentage of views for each type of video is summarized in Table I.

Next, we order the videos in each category from the most popular to the least popular, then divide each category into 1000 evenly segmented video groups and plot the distribution of their popularity in a log-log plot, to check if they follow a Zipf distribution as reported in previous studies [2]. Since different types of videos have different population sizes, we normalize them by dividing views of each type of videos over the total number of views received by that type. This is shown in Fig. 1. From this plot, we can see that each curve can be approximated to a straight line, i.e. a Zipf distribution. For News and Sports, however, the popularity falls off noticeably faster than that of the other three types.

For Fig. 1, all the videos considered for the plot have been viewed during the time window for the analysis. We take all the videos stored in the system into consideration, whether they were viewed during the measurement time window or not, and

[5]We considered two factors to choose the representative video type: the average daily views of each type, and their popularity dynamics. For the type of video with large number of views and similar popularity patterns, we chose one of them as representative. In addition, our work is a systematic solution that can also include Animes.

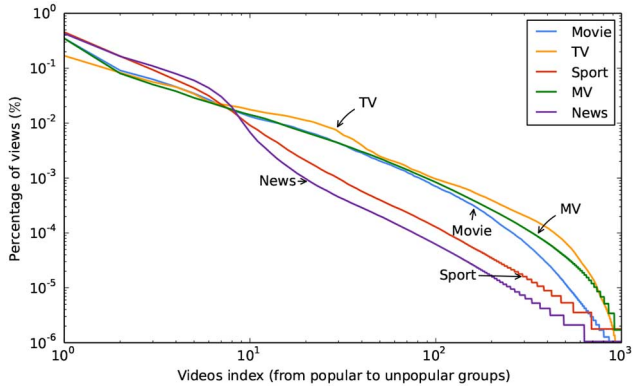

Fig. 1. Weekly video popularity distribution.

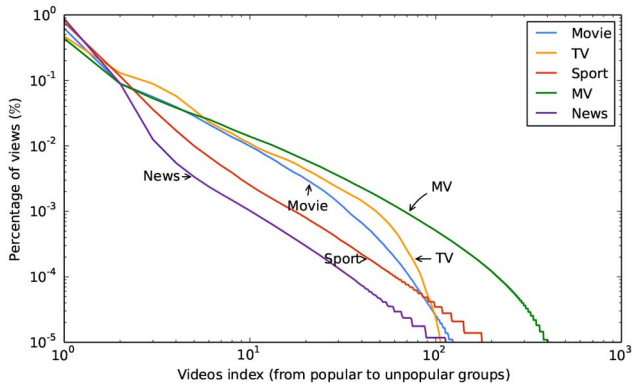

Fig. 2. Weekly video popularity distribution by involving unwatched videos.

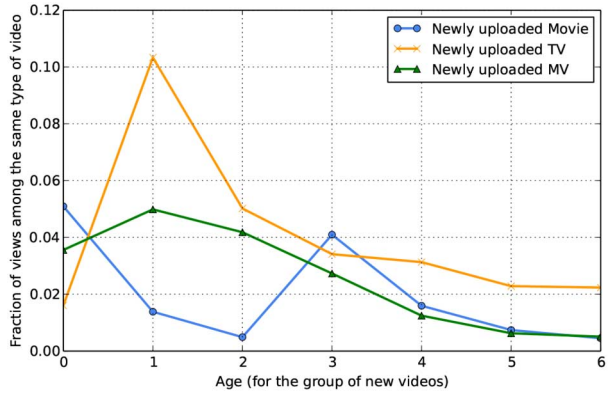

Fig. 3. Fraction of views for newly uploaded movie, TV, and MV videos.

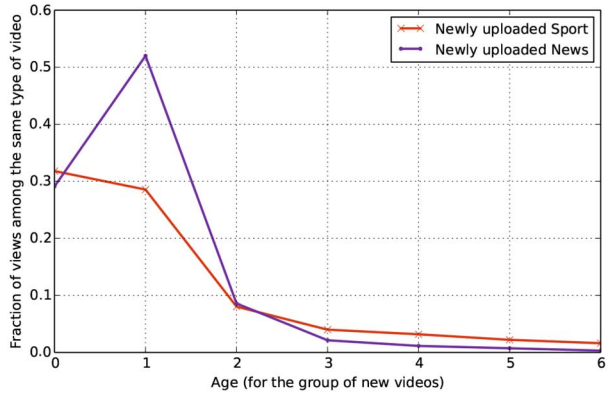

Fig. 4. Fraction of views for newly uploaded Sport and News videos.

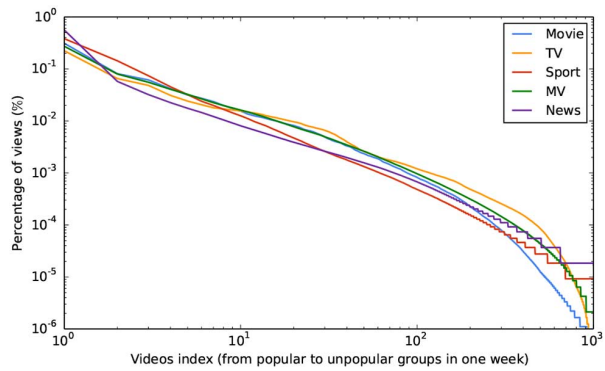

Fig. 5. Weekly video popularity distribution after removing videos with $age < 7$.

show the resultant distributions in Fig. 2. In this case, around half of MV videos were viewed, a much high percentage compared to the other types is got. It is rather surprising that even for Movies, the percentage viewed is rather small.

The results of this section indicate clearly that there is another dimension to video popularity, that is how it changes over time. We study the factors that affect the change of video popularity in the next section.

IV. Factors to Affect Video Popularity

Besides the total number of views in a time window, we are also interested in how popularity changes over time. We will study the factors that affect the dynamics of video popularity, i.e., popularity's age-sensitivity, constrained eyeball and probability of replay in this section.

A. Popularity's Age-Sensitivity

Recently updated videos are referred to as *young* videos. Generally speaking, young videos attract more views than older videos. To see which videos are age-sensitive, we gather the first 7 days' view records for newly updated videos in our database (over 7 days). Then we plot the fraction of views against age in Fig. 3 and Fig. 4.

Fraction of views is the percentage of views taken by videos updated on a particular day, compared with the total views received by that category. On the X-axis, 0 is the first day the videos went online. For News and Sports, both take about 30% of views within the first day. By the third day, the percentage of view counts of News and Sports decreased sharply. This validated our suspicion from the results in the last section that News and Sports videos are very age-sensitive. The popularity for the other three types of videos is much less age-sensitive. The cumulative percentages of views within the first three days account for a much smaller fraction of total views.

Next, we exclude all videos with age less than 7 days from Fig. 1 and plot the popularity distribution for the remaining videos (with at least some views during the 7 day period) in Fig. 5. Surprisingly, the five curves corresponding to the five types of videos are now very close to each other, and all consistent with the Zipf distribution. The sharper popularity declining rates for News and Sports as observed in Fig. 1 are no longer there.

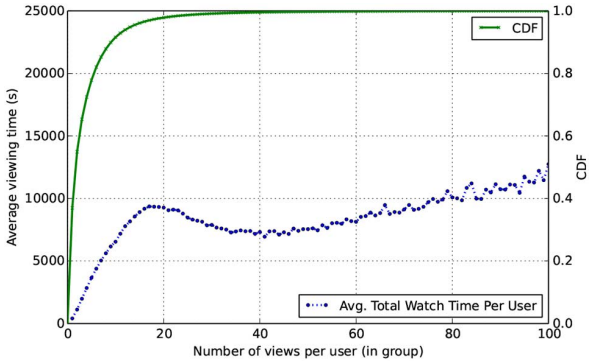

Fig. 6. Average total watch time per user in each group and the CDF for the percentage of each group.

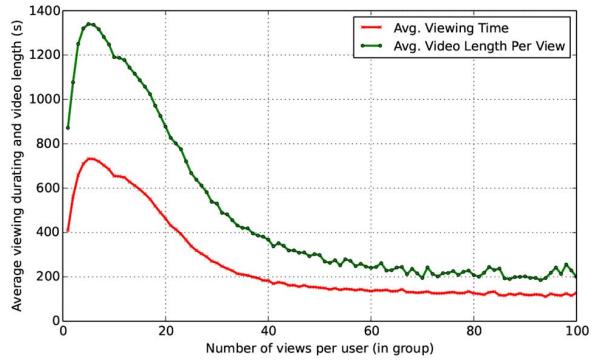

Fig. 7. Average viewing time per view for each group of users.

This leads us to suggest using the three stages to describe the video popularity of each video:

(a) the hot stage consisting of no more than 7 days;
(b) the warm stage;
(c) the cold stage when the videos are hardly viewed at all in a given day.

The behavior in Fig. 5 captures the warm stage, which is very similar for different type of videos. Those videos with no views in Fig. 2 should be in the cold stage. We conjecture that the different behaviors exhibited in the three stages are caused by different ways videos are recommended to the users (or brought to their attention). For example, in the hot stage, it is probably caused by *in-your-face* type of recommendation [4] which are afforded to only a limited subset of videos; in the warm phase, the videos are brought to users attention by slower *word-of-mouth* or *browsing-based* discovery [7], [8]. To validate these conjectures would require more in depth experimentation beyond the scope of the current paper.

B. Constrained Eyeball

What drives video popularity is users' attention, or more concretely, users' time. All the videos are competing for users' time each day. Thus, a good question is how much total time users devote to viewing online videos (on a daily basis). Does this change with increased choice of videos and number of videos viewed? In trying to answer these questions, we analyze the data and plot the distribution of the number of users and the average total viewing time per user per day against the number of views per day in Fig. 6. The X-axis is different values of daily view count. There are two curves - one is the CDF of user population (with Y label on the right-hand side of the figure); and the other is the average viewing time (with the Y label on the left-hand side of the figure). From Fig. 6, we first observe that it is rare for a user to have more than 30-40 views per day. Higher view count tends to correlate to longer viewing time, but only up to 16-17 views per day. Beyond that, the viewing time *saturates* at around the 8000 seconds level (a little over 2 hours per day).

Furthermore, in Fig. 7, we plot the average length of the video viewed by users in each group, and the average viewing time per video, against daily view count. We observe that the average length of selected videos increases with view count initially until view count equals to 5. After that, users tend to select more shorter videos (e.g. News and Sports). The average viewing time per video follows the same pattern more or less. The average viewing ratio, defined as the fraction of video viewed, remains more or less constant. Since, total eyeball, i.e., the total time spent on viewing videos, is constrained, by viewing more videos a user tends to select those videos with short duration. It is surprising that the users who viewed few videos (< 5) also choose shorter videos, even though they should be less pressed with time. We suspect that this may also be related to the type of videos selected. Perhaps people who view very few videos tend to view News, Sports or MVs rather than Movies.

C. Probability of Replay

The knowledge of daily viewing time afforded by users is not sufficient to help us sketch the dynamics of video popularity. Another key factor is how often a video is replayed by users. Replay can potentially prolong the glory days of videos. In other words, videos that users like to replay may have a more stable popularity compared with those videos rarely replayed by users.

It is not obvious how best to quantify a user's replay activity. We first consider the extent of replay (out of total views) from a user perspective for each type of videos. Each user's views are classified into different types before we do the following calculation for each user. Let v_i denote the total number of views for user i attributed to a particular type of videos.[6] We use $\vec{v_i} = (v_{i1}, \ldots, v_{iM})$ to represent the view distribution for user i, referred to as *view pattern*. Here v_{ij} is the number of views of user i for video j. Naturally, $\sum_j v_{ij} = v_i$. With this representation, we can calculate the replay percentage for user i as

$$\eta_i = \frac{\sum_{j=1}^{M}(v_{ij} - 1) \cdot I(v_{ij} > 0)}{v_i}. \quad (1)$$

$I(v_{ij} > 0)$ is the indicator function with value 1 if $v_{ij} > 0$ and otherwise. denotes the total number of videos of some category. Percentage of repeated views could be used to reflect user replay behavior partially. The percentage of replayed videos is calculated for each user according to EQ. 1 within one week. In Tencent Video, most users have their social network identities, i.e., Tencent QQ numbers. Users can login in Tencent Video with QQ numbers before they view videos. For these users, we can exactly identify each of them and track their view records to

[6]Note, each type of videos is treated separately, so we only need to discuss views of a particular type.

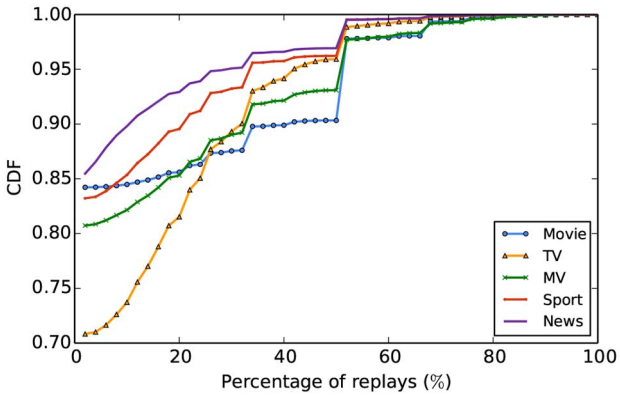

Fig. 8. CDF for the percentage of replays.

study the replay behavior. The CDF curve of replay percentage for all tracked users is plotted in Fig. 8. The X-axis represents the replay percentage while the Y-axis represents the percentage of users whose total replay percentage is no more than the value given by X-axis. As we can see, the replay percentage is very low, especially for News - more than 90% users have replay percentage less 10% than. We can conclude that most users seldom go back to play a video once already viewed.

However, the simple replay percentage does not tell us how the replayed views are distributed. Take the case with two videos and two users as an example, if the first user's view pattern is $\vec{v}_1 = (2, 2)$ while the second user's view pattern is $\vec{v}_2 = (3, 1)$, the replay percentage is 50% for both. So replay percentage alone cannot distinguish them. To more distinguish users with the same replay percentage but different replayed view distributions, we borrow the idea of entropy and define *replay entropy* to capture the distribution of the replayed views. The replay entropy of user i is

$$H_i = -\sum_{j:v_{ij}\neq 0} \frac{v_{ij}}{v_i} \ln \frac{v_{ij}}{v_i}. \tag{2}$$

v_i is the total number of views of user i attributed to some video type over a fixed period.[7] Because the number of views by a single user is much less than the entire video population, $v_{ij} = 0$ is the most common case. For replay entropy, only those videos with $v_{ij} > 0$ are included. Note, for different periods, v_i and $\vec{v}_i$ will be different. For example, daily (weekly) entropy means the views used to calculate replay entropy are collected over a single day (week). But we just simply use H_i to denote replay entropy for user i if v_i and $\vec{v}_i$ are given and the time period is irrelevant.

The use of entropy has been considered in many previous works. For example, [9] defined similar view entropy for IP addresses and videos. The objective is to detect those videos that are attacked by fake views and the IP addresses that are likely to be generating fake views. In [10], the authors defined view entropy for each video to observe how the video's views distribute in different regions. The video entropy will be lower if the view distribution is more concentrated. Different from them, we define entropy for each user to study the normal users' replay behavior.

Intuitively speaking, according to the property of entropy function, a lower entropy value for a user implies a higher probability for that user to replay (some) video. It also helps us differentiate users with the same percentage of replayed views. For example, the view pattern $\vec{v}_1 = (3, 1)$ will have a lower entropy value compared with $\vec{v}_2 = (2, 2)$. We derive a few additional properties of replay entropy as follows.

For any user i, his views can be ranked by decreasing order, without loss of generality. If the video population is denoted by M and user i's total view count is v_i, then his replay pattern can be denoted by a vector $\vec{v}_i = (v_{i1}, v_{i2}, \ldots, v_{iM})$, with $v_{i1} \geq v_{i2} \geq \ldots \geq v_{iM}$ and $\sum_{j=1}^{M} v_{ij} = v_i$. The replay entropy is determined by the replay pattern. More skewed replay pattern, defined later, will result in lower entropy.

Proposition 1: The maximum replay entropy achieved by a user with total v_i views is $\ln v_i$ if $v_i \leq M$ or $\ln M$ if $v_i > M$.

Proof: The proof is straightforward. According to the property of the entropy function, it achieves maximum value when $v_{i1} = v_{i2} = \ldots = v_{iM}$. If $v_i < M$, $v_{ij} = 1, \forall j$, then the replay entropy is $\ln v_i$. The case with $v_i > M$ can be derived in a similar way. ■

For a single user, it is impossible for $v_i > M$, thus we can conclude that the replay entropy is no more than $\ln v_i$.

For a user i with replay pattern $\vec{v}_i$, we define a skew-increasing move operation if a view is moved from a video with smaller view count to another video with higher view count. In other words, for a pair of videos x and y with $v_{ix} > v_{iy}$, the result of a skew-increasing move operation is $v_{ix} + 1$ and $v_{iy} - 1$.

Proposition 2: For two users i and k with the same number of total views v, if replay pattern of user i can be reached from replay pattern of user k with finite number of skew-increasing move operations, the replay entropy of user i is less than that of user k.

The detailed proof is presented in Appendix. Let us intuitively explain this proposition. Since the total number of views is the same for both users and view counts v_{ij} are ranked by decreasing order, skewer replay pattern implies if $v_{ix} > v_{kx}$, there must exist $v_{iy} < v_{ky}$ for some $y > x$. The replay pattern of user i is more concentrated than that of user k, resulting in smaller entropy value. In this way, we can distinguish users with the same replay percentage.

We can compare users' replay entropy to judge their replay behavior given fixed total views. However, it is unfair to compare replay entropy directly if users have different total number of views. The users with more total views tend to have larger replay entropy value than the users with smaller total views since more views could involve more videos. Once any additional video is involved, the entropy will be enlarged [9]. We track all users in our system for one week and computed their weekly entropy for News, and plot the result in Fig. 9. The X-axis is the number of views, Y-axis is the value of entropy and each point represents one user. There is a straight line $y = \ln x$ as the upper bound of replay entropy. A user tends to have fewer replay views if his entropy is closer to the upper bound. Understandably, there is a spectrum of different user replay behaviors. One observed result is that the users with higher view counts

[7]This slight abuse of notations in v_i and v_{ij} helps to save symbols.

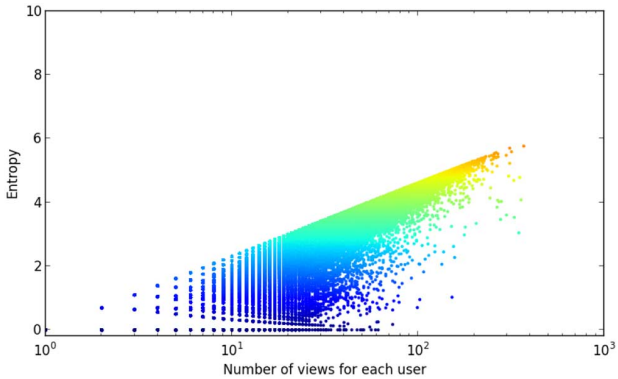

Fig. 9. Replay entropy for user views of News.

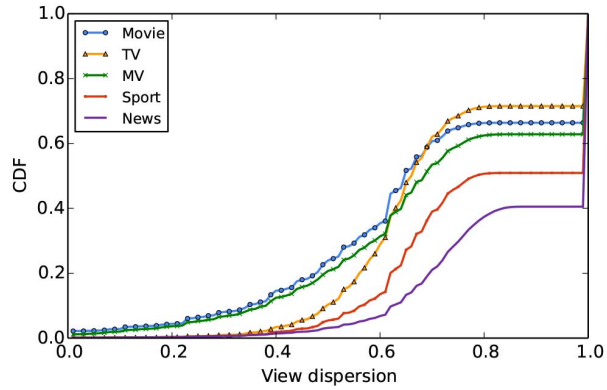

Fig. 10. CDF of view dispersion for each video type.

tend to have lower replay probability for their views. We have also tracked the replay entropy for the other video types. The results are similar, but due to limited space those results are omitted. From Fig. 9, the users with more total views tend to have higher entropy value, even though their replay percentage may be larger.

We derive bounds on replay entropy in the following proposition.

Proposition 3: If a user i has total $v_i < M$ views, of which $\eta_i v_i$ are replayed views, approximately the maximum value of replay entropy is

$$H_i^{max} = \ln(1-\eta_i)v_i.$$

The minimum value of replay entropy is

$$H_i^{min} = -\left(\eta_i + \frac{1}{v_i}\right)\ln\left(\eta_i + \frac{1}{v_i}\right) + \left(1-\eta_i - \frac{1}{v_i}\right)\ln v_i.$$

Proof: According to Proposition 1, replay entropy reaches maximum value if these views distribute as even as possible. Since there are $\eta_i v_i$ replayed views, on average a video is played for $\frac{1}{1-\eta_i}$ times. The entropy is $H_i^{max} = -\sum_{j=1}^{(1-\eta_i)v_i} \frac{1}{(1-\eta_i)v_i} \ln \frac{1}{(1-\eta_i)v_i} = \ln(1-\eta_i)v_i$.

Similarly, the minimum value is reached if $\eta_i v_i$ views are for a single video and the rest views are for different videos. Then, $H_i^{min} = -(\eta_i + \frac{1}{v_i})\ln(\eta_i + \frac{1}{v_i}) + (1-\eta_i - \frac{1}{v_i})\ln v_i$. ■

We are finally ready to show how the replay views distribute, in terms of normalized replay entropy for each user, which allows us to compare replay frequency of users with different view counts. The normalized entropy is defined as

$$H_i^n = \begin{cases} 1, & v_i = 1 \\ \frac{H_i}{\ln v_i}, & \text{otherwise.} \end{cases} \tag{3}$$

Proposition 4: If a user's percentage of replayed views is η_i, the normalized replay entropy is in the range $[1-\eta_i - \frac{1}{v_i}, 1]$.

Proof: According to Proposition 3, the maximum normalized entropy is

$$\frac{H_i^{max}}{\ln v_i} = 1 - \frac{-\ln(1-\eta_i)}{\ln v_i} \le 1.$$

Since $\eta_i + \frac{1}{v_i} \le 1$, the minimum normalized entropy is

$$\frac{H_i^{min}}{\ln v_i} = 1 - \eta_i - \frac{1}{v_i} - \frac{\left(\eta_i + \frac{1}{v_i}\right)\ln\left(\eta_i + \frac{1}{v_i}\right)}{\ln v_i} \ge 1 - \eta_i - \frac{1}{v_i}.$$

The maximum value 1 is achieved if v_i views are for different videos; while the minimum value $1-\eta_i - \frac{1}{v_i}$ is achieved if the user repeatedly view a particular video for $\eta_i v_i + 1$ times. ■

Based on the range of normalized replay entropy value, we introduce *view dispersion* by rescaling normalized entropy for user i [10], as

$$D_i = \frac{H_i^n - \left(1-\eta_i - \frac{1}{v_i}\right)}{\eta_i + \frac{1}{v_i}}. \tag{4}$$

The nice property of *view dispersion*, D_i, is that it is always in the range [0.1]. Higher D_i implies more uniformly distributed view pattern. Its value is mainly determined by how the replayed views distribute. To better understand the relationship between D_i and $\vec{v}_i$, let us consider an example with three users. The view patterns are $\vec{v}_1 = (2,2,2,1,1)$, $\vec{v}_2 = (4,1,1,1,1)$ and $\vec{v}_3 = (2,2,2,2)$ respectively. The replay percentages are $\eta_1 = \eta_2 = 0.375$ and $\eta_3 = 0.5$. Although user 3 has the highest replay percentage, his views are evenly distributed among four videos, so its viewer dispersion is expected to be higher than user 2, whose view pattern is very concentrated. From simple calculation, the replay entropies are $H_1 = \frac{3}{4}\ln 4 + \frac{1}{4}\ln 8$, $H_2 = \frac{1}{2}\ln 2 + \frac{1}{2}\ln 8$ and $H_3 = \ln 4$. With replay entropy and replay percentage, we can find the view dispersions are $D_1 = 0.5$, $D_2 = 0.25$ and $D_3 \approx 0.47$. The result is consistent with our expectation. D_3 is comparable with D_1 since their view patterns are similar. D_2 is the lowest one since user 2's view pattern is very skewed and his replays are concentrated on a single video.

We plot the CDF of view dispersion for each video type in Fig. 10. This time we only track those active users with weekly view more than 5 to observe their replay behavior. X-axis is the view dispersion while the Y-axis is the percentage of users with dispersion less than the value given by X-axis. It is not difficult to interpret the results in Fig. 10. The News and Sports have the lowest replay probability. About 60% and 50% users view News and Sports without any replay over one week time. For Movie, TV, and MV, the low view dispersion implies more concentrated view pattern, that is on average the replayed times per video is more than that of News and Sports. Even so, some fraction of users view videos with repetition, the fraction of users with extremely low view dispersion is very few, which implies that users seldom replay a video for many times. Instead, many different videos may be replayed by a user, each for a few times.

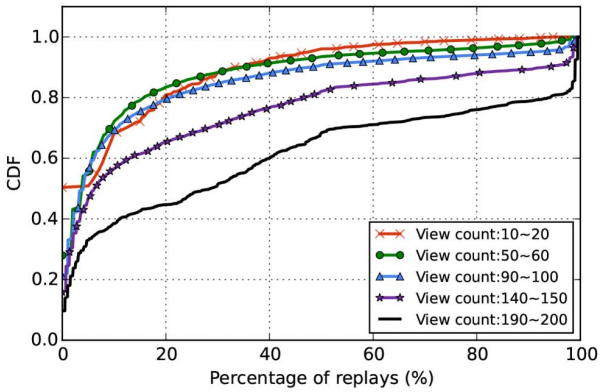

Fig. 11. CDF curves of replay percentage given the range of viewed videos.

It is more likely that these videos are replayed because of poor QoE or incomplete view from earlier play.

In order to better understand users' replay behavior, we study the distribution of replay percentage given the number of different videos viewed during a day. Let w_i represent the number of different videos viewed by user i. Then, if we have enough user view records, we can classify users into different sets based on w_i and calculate the percentage of replays for each user in each set. After that, we can plot the CDF curve of the replay percentage for each set. However, the fact is that the view records are not enough for those sets with active users who view many different videos each day. These users only take a small fraction of total users and there is no enough samples to calculate distribution of replay percentage precisely. To enrich the samples within each set, we classify users for every ten different viewed videos. For example, users with $10 < w_i \leq 20$ are classified into one aggregated set. Then, we plot the CDF curves of the replay percentage for each aggregated set in Fig. 11. Each curve represents the replay percentage distribution of one aggregated set. As we can observe, with more different videos viewed, users have higher replay percentage, implying that active users replay more videos than inactive users.

According to the results shown in Fig. 11, we can draw the conclusion on replay behavior as follows: (a) the overall probability to replay a video is very low; (b) the replays of News and Sports are less concentrated than the other three types; (c) there is almost certainly no hot video to attract users to replay again and again; (d) those active users who view many different videos per day tend to replay videos with higher percentage. We believe that the real frequency to replay a video for enjoyment is even lower than what our result shows because users may replay videos due to other reasons, such as poor QoE or incomplete views. One implication of low replay probability is that video popularity is expected to decrease with time from its peak value gradually, since most users never go back once the videos have been viewed.

D. Discussion

In summary, here are the key observations about the dynamics of video popularity.

- The dynamics of a video popularity depends on its type. The popularity of News and Sports tend to be more age-sensitive, which decays faster with video age. But all videos go through three stages: hot, warm and cold. For most videos, the hot stage lasts less than 7 days. The length of the warm stage can vary widely depending on the video type.
- Users' viewing time is the source of videos' lifetime. Users have limited time resource that does not always increase linearly with the number of videos played. With more view count per day, users tend to select videos with shorter duration instead of shortening view ratio.
- A video is rarely replayed. This means video delivered by VoD is rather like multicast happening in slow motion; the step of delivering to users is by pull, which happens asynchronously.

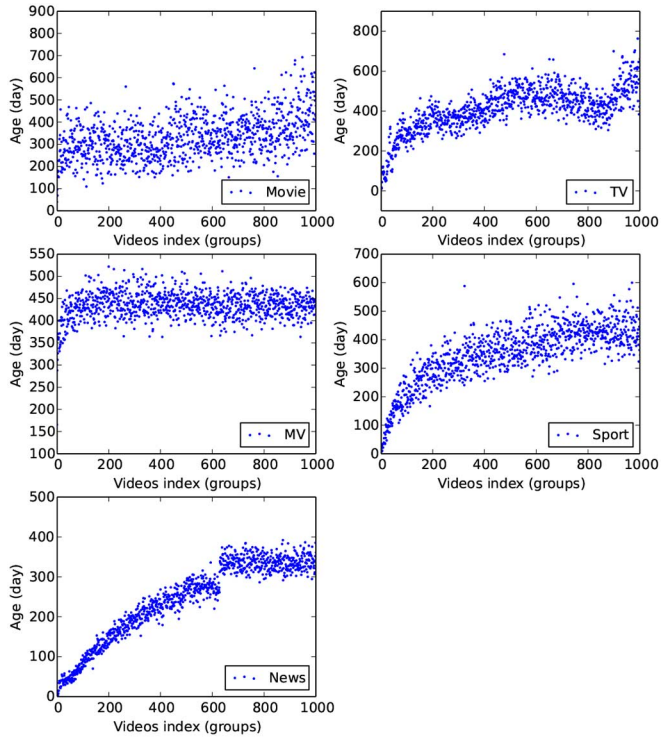

Fig. 12. Age dynamic for different types of video.

Because of the dynamic perspective of video popularity, video popularity is predictable by age to different extents, depending on the type of the video. In Fig. 12, we plot videos (divided into 1000 groups) ordered by (weekly) popularity against age. We can observe that for News and Sports, the relative popularity of a video is strongly dependent on its age; whereas MVs' popularity is much less sensitive to their age. The situations with Movie and TV episodes are between News and MV. This insight is very helpful for us to propose a simple mixed replication strategy (of LFU and FIFO) in the next section.

V. Video Caching Strategies

The better understanding of video popularity dynamics is very helpful to improve service quality for online videos. It is well known that the current VoD service relies heavily on CDN servers for content distribution. CDN servers are usually equipped with certain storage for video caching. If the videos

to be distributed are cached by CDN servers, not only bandwidth resource is saved but also latency and response time are shortened. Since the cache size is limited compared with the size of the whole video set, it is essential to place the right videos on CDN servers. Intuitively speaking, the dying videos and the videos that cannot attract user interests should not be heavily replicated on the CDN servers. With the knowledge of the dynamics of video popularity, we can enhance the caching efficiency. In this section, we first discuss some common video caching strategies; then we propose the design of a mixed video caching strategy based on measurement study. Finally, through trace-driven simulation, we compare the mixed strategy with the other strategies to show its advantages.

A. Design Principle

With limited cache storage, a good caching strategy should keep those videos that will be requested most in the near future. For video caching in CDN, there are many well known algorithms. In our work, we consider two of them, i.e., Least-Frequently Used (LFU) algorithm [11] and Least-Recently Used (LRU) algorithm. Briefly speaking, LFU records how often a video is requested during the latest time window.[8] The window duration is a parameter that can be tuned. Each time if there is a newly arrival video, CDN servers always discard the video received least requests during last time window first. Intuitively speaking, if the video popularity is static and the time window is long enough, LFU can find the most popular videos quite precisely. Similar to LFU, LRU will cache those videos that are most recently requested. LRU maintains a flag for each cached video to indicate how long the video is in idle state without any request. Once a non-cached video is requested, it will be stored to replace the video with longest idle state.

Unfortunately, in the real world video popularity is highly dynamic, especially for News and Sports. The LFU and LRU algorithms cannot predict the popularity of newly updated videos or videos with drastic popularity changes. Based on our measurement, once a video loses its attraction to users, its popularity will diminish and rarely go back. In other words, there exist videos that will become very popular, but cannot be captured by LFU strategy very well. According to our study, it is highly possible that those videos are newly updated without many view records. Thus, a natural idea is to cache new videos on CDN servers, i.e., the First-in-First-out (FIFO) algorithm. By using the FIFO, the oldest videos are always discarded first for video replication.

LFU can detect popular videos with relatively stable popularity; while FIFO can cache popular videos with young ages or videos with little view history. The right way should cache both kinds of popular videos, so we propose a simple mixed caching strategy. For each video j, we maintain both its age a_j and its view count c_j in the last time window. Each time when we need to discard a video, with probability p the oldest video is discarded, and with probability $1-p$ the least frequently requested video is discarded. If the cache capacity is C, by running the mixed strategy, about $C \times p$ storage is allocated to execute FIFO strategy and about $C \times (1-p)$ storage is allocated to execute LFU strategy.

[8]We assume that video replacement strategies are executed on daily basis, since most VoD systems update cached videos on CDN servers during light load period in each day, e.g., early morning.

In our simulation, we implement an approximated version. We divide the cache space into two parts with size $C \times (p-1)$ and $C \times p$ respectively. Videos with most history views are selected to be cached in the first part until it is full. Then, the videos not cached yet with young age will be selected to cache in the second part. Let s_j denote the size of video j, then the detailed algorithm is shown in Algorithm 1.

Note, in our work both LFU and LRU are studied through trace-driven simulation, and LFU alrgorithm outperforms LRU algorithm. Thus, for designing the mixed video replacement strategy, we use the combination of LFU and FIFO. Our study does not assert that the LFU algorithm always outperforms the LRU algorithm. We can easily extend to mixed algorithm by combing LRU with FIFO, or LRU with FIFO and LFU. Actually, the design space is quite broad. Owing to limited space, we only focus on the mixture of LFU and FIFO, and show its advantages. The design of more complicated mixed caching algorithms is left to our future work.

As a performance benchmark, we implement the offline algorithm, which assumes all views in the future are known. In simulation, since the future views are known, the offline algorithm can always cache the most popular videos. We use the hit rate as the metrics, which is defined as

$$\text{Hit rate} = \frac{\text{Number of requests for cached videos}}{\text{Total number of requests}}.$$

Algorithm 1: Mixed Strategy for Video Replacement.

Data: Given a period T, update view count list (j, c_j) and age list (j, a_j)
Result: $\mathcal{S}$: the set of cached videos
Initialization: sorting (j, c_j) based on c_j descendingly cache size = 0
while: *cache size* $\leq (1-p) \times C$ **do**
  for *sorted* (j, c_j) **do**
    caching video j in $\mathcal{S}$
    cache size = s_j+ cache size;
Initialization: sorting (j, a_j) based on a_j increasingly cache size = 0
for (j, a_j) **do**
  if $j \notin \mathcal{S}$ and *cache size* $\leq p \times C$ **then**
    caching video j in $\mathcal{S}$
    cache size = s_j+ cache size;
  else
    skip video j

Hit rate is the objective we try to maximize. It is obvious that the offline strategy can achieve the highest hit rate with limited storage since the number of requests for each video in the future is known in advance.

Note, based on the measurement study, p should be different for different types of videos to achieve high hit rate. For example, the p of News should be very close to 1 since most News are very age-sensitive. In the mixed strategy, though no explicit

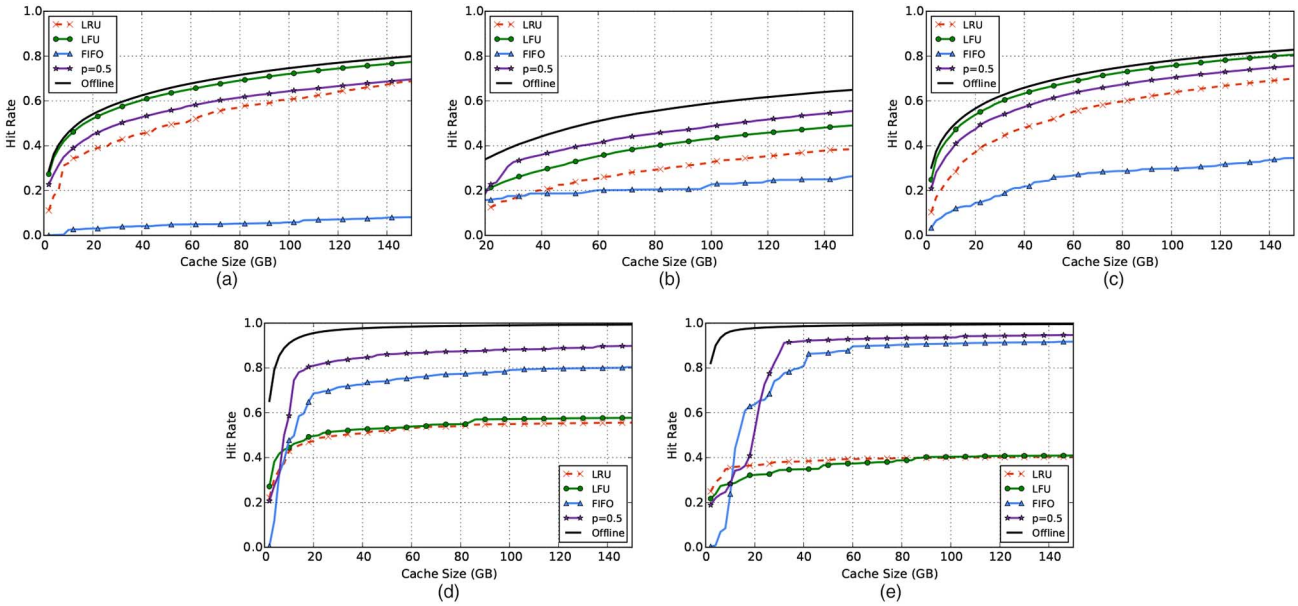

Fig. 13. Caching performance for different types of videos. (a) Movie. (b) TV. (c) MV. (d) Sport. (e) News.

p is defined for each type of videos, it is not difficult to determine a suitable p for each type of videos based on historical view records. We can search a p achieving the highest hit rate from each type of videos' view records and use that in the corresponding mixed strategy in the future.

B. Performance Evaluation

We check the effectiveness of the mixed strategy by using the trace data collected from Tencent Video to conduct simulation. The trace includes view counts for each video for each day that can drive the user views in our simulation. The number of views is counted on daily basis, and the video replacement is also executed on daily basis. The offline algorithm can always cache the videos that will be viewed most in the next day. The total video catalog is huge, so we only assume a small fraction can be cached. For each type of videos, part of cache space is used for each type of videos. For simplicity, we assume that the cache size is no more than 100 GB for each type. Note, how to decide the storage size on each server and optimize the storage allocated to each type of videos are also important problems worth studying, but they are orthogonal to our work and not covered in this paper. We focus on how to fulfill the most user requests by replicating the right videos with limited storage.

For the first simulation, we run the mixed strategy by letting $p = 0.5$ and compare its hit rate with those of LFU, FIFO and offline. The cache storage varies from 20 GB to 100 GB The simulation results are presented in Fig. 13. As we can see, for Movie, TV and MV, the hit rate of LFU algorithm is the best one and very close to the hit rate of the offline algorithm. For Sports and News, the optimal caching strategy is the mixed strategy. For news, the FIFO algorithms also outperforms the LFU algorithm. The results are consistent with the measurement results in Fig. 12. News is very age-sensitive and rarely replayed by the same user, thus the FIFO algorithm can fulfill the most popular News given enough cache. For Movies, TV and MV, they are insensitive to age. For these videos, on the one hand, users are not in a hurry to view the new updated ones; on the other hand, users may replay them with higher probability as shown in the last section. Since popularity does not change drastically with time and higher replay probability, the LFU algorithm can fulfill the most of the popular ones. Sports is a very special type since neither LFU nor FIFO can achieve good performance. That means some sport videos' popularity is very age-sensitive while some sport videos last popular for quite a long time. The mixed strategy can cover both cases so as to achieve the best performance.

In Fig. 13 with Movie, TV and MV, the best caching algorithm is LFU, a special case of the mixed strategy with $p = 0$. A natural question is what is the optimal value of p to achieve the best hit rate. Actually, unless we know exactly how sensitive to age the video popularity is and how long a popular video can last, it is difficult to give an optimal p. Instead of searching the optimal p, we let $p = 0.5$ that takes care of both time sensitive videos and popularity stable videos equally. Although, storage is wasted a little bit, we can ensure popular videos are cached. We compare different p for the Sport and News videos. As shown in Fig. 14, we compare several mixed strategies with different p. Though $p = 0.5$ is not the best one, its hit rate keeps increasing and getting closer to the optimal solution with increasing storage size. This phenomenon can also be observed in Fig. 13 with Movie, TV and MV. Though the curve with $p = 0.5$ is not optimal, the hit rate keeps increasing with storage size.

Even though, we have evaluated the cache algorithms separately for each category of videos, in practical systems we have to consider how to allocate the cache storage to each video category, because the storage allocation also affects the video hit rate. Thus, we introduce a simple storage allocation strategy with the goal maximizing overall hit rate such that the mixture

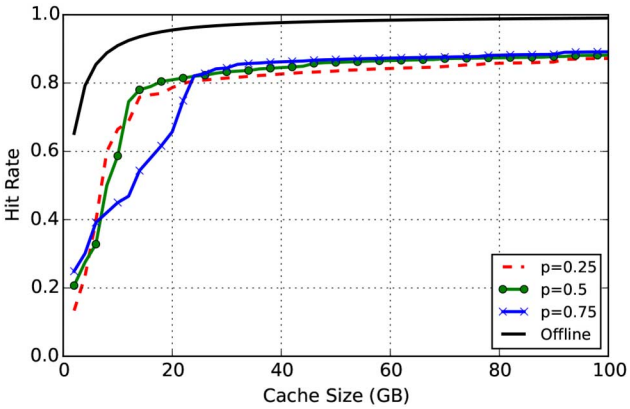

Fig. 14. Different value of p for Sport.

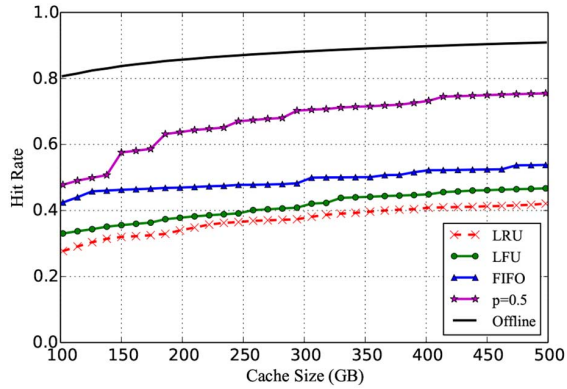

Fig. 15. Simulation of all video categories.

strategy can be applied to practical systems. First of all, the objective of storage allocation strategy is to maximize the overall hit rate, which is defined as the user views for all cached videos divided by all views. As we have shown through simulation, for each video category, the hit rate curves of all algorithms monotonically increase with allocated storage size. We can take the hit rate as a function $h_k(L_k)$, where L_k represents the storage size allocated to category k for video caching. Note, the knowledge of $h_k(L_k)$ is very essential for us to derive the optimal storage allocation. We conduct trace-driven simulations without assuming any implicit user arrival or departure patterns, though it is difficult to derive the expression of $h_k(L_k)$ exactly. We just assume that $h_k(L_k)$ is a concave function that monotonically increases with L_k. Let α_k be the fraction of views taken by video category k and K be the total number of video categories. It is not difficult to verify that the overall hit rate is equal to $\sum_{k=1} K\alpha_k h_k(L_k)$. Then, maximizing overall hit rate is equivalent to the following optimization problem:

$$\begin{aligned} \min \quad & -\frac{1}{K}\sum_{k=1}^{K}\alpha_k h_k(L_k,) \\ s.t. \quad & \sum_{k=1}^{K} L_k = L. \end{aligned} \tag{5}$$

Here, L is the total available storage. Since $h_k(L_k)$ is a concave function, $-h_k(L_k)$ is a convex function and the maximized hit rate is achieved when $h_1'(L_1) = \ldots = h_K'(L_K)$ with $\sum_{k=1}^{K} L_k = L$.

With the insights obtained from above analysis, we propose a simple storage allocation strategy, though we do not know the expressions of $h_k(L_k)$. For the simulation results in Fig. 13, the incremental storage is 2 GB for each simulated point. We use the simulated incremental hit rate as the approximation of $\Delta_k = h_k(L_k + 2) - h_k(L_k)$. Given total L GB storage, we only need to do $\frac{L}{2}$ times storage allocation. For each allocation, 2 GB storage will be assigned to the video category with largest $\alpha_k \times \Delta_k$ and L_k is increased by, 2 GB correspondingly. The detailed algorithm is presented in Alg. 2.

Algorithm 2: Storage Allocation Strategy for Each Video Replacement Strategy.

Data: $\alpha_1, \ldots, \alpha_K$, the fraction of views taken by each video category
Result: $L_1, \ldots, L_K$: the solution of storage allocation
Initialization:
for $k = 0, k < K, k++$ **do**
  $L_k = 0$
for $i = 0, i < \frac{L}{2}, i++$ **do**
  calculate $\Delta_k = h_{L_k+2} - h_{L_k}$ for $k = 1, \ldots, K$
  choose k' with largest $\alpha_k \times \Delta_k$
  $L_{k'} = L_{k'} + 2$

With the storage allocated by Alg. 5.2, we conduct the trace-driven simulation again with all video categories. The result is shown in Fig. 15. Note, for different video replacement strategies, they have different optimal storage allocation solutions. We compare the hit rates for different video replacement strategies with optimal storage allocation for each of them. As we can see, the mixed strategy proposed by us is the best one that significantly outperform the other algorithms.

Compared with the simple video replacement algorithms, such as LFU and LRU, the mixed strategy does cost more computation overhead for implementation. However, we believe that it is still worth implementing the mixed algorithm since required additional computation overhead is moderate. Implementing the mixed strategy only needs the knowledge of parameters p, α_k and L_k for each video category. On one hand, the number of video categories is limited, with no more than dozens. On the other hand, it is not necessary to update these parameters frequently. They can be obtained by offline test through the framework proposed in our work once every week or month. In real systems, there may exist dozens of video types and information collection is challenging. Exactly optimizing the storage allocation may be difficult. How to achieve high

TABLE II
LIST OF NOTATIONS USED IN THIS PAPER

Notation	Explanation
v_i	the total number of views for user i attributed to a particular type of videos
v_{ij}	the number of views of user i for video j
η_i	the replay percentage for user i
H_i	the replay entropy of user i
H_i^n	the normalized entropy of user i
D_i	the view dispersion for user i
a_j	the age of video j
c_j	the history view count of video j
p	the probability with which the oldest video is discarded
C	the cache capacity

overall hit rate in this case is our future work. A list of notations used in this paper can be found in Table II in Appendix.

VI. RELATED WORK

Over the last few years, several works have been devoted to the study of the popularity and other statistical properties of online videos. For example, Cha *et al.* [5] collected traces from two large UGC video websites, and analyzed their popularity distribution/evolution as well as the influence of content duplication. In [10], authors observed the relationship between popularity and locality of YouTube videos. They showed social sharing widened the geographic reach of a video and affected its popularity correspondingly. However, we focused on studying the popularity of non-UGC videos, purchased and made available by content providers.

The study of popularity is not restricted to videos. In [12], Szabo and Huberman investigated how to predict the future popularity based on the understanding of early popularity of Digg's submission and YouTube's video respectively. By exploring a massive dataset of Twitter, Yang and Leskovec studied temporal patterns associated with online content in [13] and presented a model to predict tweet popularity. Our work has the same ultimate goal. Our approach is to study the dynamics properties of video popularity and use them as basis to predict video popularity dynamically.

Many previous works showed and repeatedly verified that content popularity follows a power-law distribution, e.g., Jacob *et al.* examined the popularity of Wikipedia topics and Web pages with distributions characterized by fat tail in [14]. Some past work also concluded that some popularity evolution follows exponential distribution [15], based on data from a Dutch online TV portal, and historical data on DVD rentals in the US. In [16], it was found that during online video browsing, the video session length distribution is related to power-law distribution. In comparison, our work in this paper goes beyond a static model of video popularity, and presents observations on the replay viewing behavior. Also, [17] studied a large VoD system with more than 150,000 users, which is deployed by China Telecom. The work covered user behavior, content access patterns, and their implications for system design. The authors also proposed a more accurate modified Poisson distribution to analyze user arrival rate and found that video session lengths had a weak inverse correlation with the videos popularity.

Entropy is an effective method to evaluate the frequency how users replay a certain video. In our previous work [9], entropy is used to detect those IP addresses generating fake views or videos that are attacked by fake views. Fake view refers to the video views generated by machine or programs with the purpose to boost a video view count to attract user eyeballs. Thus, for fake view generators, their view entropy is quite low since the boosted video receives overwhelmed replays. A machine learning model together with the video age and video type is designed to detect fake view generators and the videos attacked by fake views. In this work, we use the technique introduced in [9] to exclude fake views first and apply the entropy technique for tracked users to study the normal user replay behavior.

There exist works discussing the user sharing and recommendation effects on the video views. In [4], Zhou *et al.* performed a measurement study on data sets crawled from YouTube and recognized that there is a strong correlation between the view count of a video and the average view count of its top referred videos. The work [18] characterized video-based interactions that emerged from YouTube's video response feature, and identified the video response view and the interaction network view. [19] can predict video view counts with high accurate based on users' synchronous sharing behavior.

The caching strategy on CDN has been studied for Web applications in [20], [21]. In [22] the authors demonstrated that certain cache networks were non-ergodic in that their steady-state characterization depended on the initial state of the system. Then, the authors established conditions for a cache network to be ergodic. In [23], the authors proposed a unified methodology to analyse the performance of caches by extending and generalizing Che's approximation. Several caching policies were considered by taking into account the effects of temporal locality. Simulation and trace-driven experiments were conducted to validate their results. In [24], the authors defined a more realistic arrival process for the content requests generated by users with the aid of YouTube. A new parsimonious traffic model, i.e., Shot Noise Model (SNM), was proposed that enables users to natively capture the dynamics of content popularity. Different from their works, our work focuses on the user behavior, dynamic changes of video popularity and their implications for video cache replacement strategies.

VII. CONCLUSION

In this work, based on the big data available from a real-world VoD system, we studied video popularity as a dynamic system. We found that as a percentage of total views, replays of online videos are not significant. So one way to think of video popularity can be based on view count, or the number of users reached eventually. At any particular time, however, the dynamics of video popularity depends on the age of the video, following a pattern for each type of video. The relative popularity of different videos at a given time is complex, and is governed by many factors. According to these observations, we propose a mixed strategy to determine the videos cached on CDN servers. Since the mixed strategy takes both age-sensitive videos and popularity stable videos into account, most popular videos can be captured to achieve high hit rate.

APPENDIX

Proof of Proposition 2: We study the new entropy value after one skew-increasing move operation by moving a view from a

video y to a video x with $x < y$ and $v_{ix} > v_{iy}$. The new replay pattern is $\vec{v}_i^{'} = (v_{i1}, \ldots, v_{ix}+1, \ldots, v_{iy}-1, \ldots, v_{iM})$. The new replay entropy is

$$H_i' = H_i - \frac{v_{ix}+1}{v_i}\ln\frac{v_{ix}+1}{v_i} - \frac{v_{iy}-1}{v_i}\ln\frac{v_{iy}-1}{v_i} + \frac{v_{ix}}{v_i}\ln\frac{v_{ix}}{v_i} + \frac{v_{iy}}{v_i}\ln\frac{v_{iy}}{v_i}.$$

Let $\Delta = H_i^{'} - H_i$ denote the change of entropy value, then

$$\begin{aligned}\Delta &= \frac{1}{v_i}(-(v_{ix}+1)\ln(v_{ix}+1) + (v_{ix}+1)\ln v_i \\ &\quad - (v_{iy}-1)\ln(v_{iy}-1) + (v_{iy}-1)\ln v_i + v_{ix}\ln v_{ix} \\ &\quad - v_{ix}\ln v_i + v_{iy}\ln v_{iy} - v_{iy}\ln v_i) \\ &= \frac{1}{v_i}(v_{iy}\ln v_{iy} - (v_{iy}-1)\ln(v_{iy}-1)) \\ &\quad - \frac{1}{v_i}((v_{ix}+1)\ln(v_{ix}+1) - v_{ix}\ln v_{ix})\end{aligned}$$

Consider the function $f(x) = x\ln x$ with $f^{'}(x) = \ln x + 1 > 0$ and $f^{''}(x) = \frac{1}{x} > 0$ as $x > 0$, $f(x)$ is monotonic increase convex function. Then, $f(v_{ix}+1) - f(v_{ix}) > f(v_{iy}) - f(v_{iy}-1)$ and $\Delta < 0$. That means the replay entropy will decrease as long as a view is moved from a video with less view count to a video with more view count.

For the general case of two users with $\vec{v}_i = (v_{i1}, \ldots, v_{iM})$ and $\vec{v}_k = (v_{k1}, \ldots, v_{kM})$, if the replay pattern of user k can be transferred to the replay pattern of user i by finite steps of skew-increasing move operations, then $H_i < H_k$. ■

ACKNOWLEDGMENT

The authors would like to thank Y. Hua, F. Liang, and S. Yao from Tencent Video for assisting the generous data support, as well as for their discussions and suggestions.

REFERENCES

[1] Y. Chen, B. Zhang, Y. Liu, and W. Zhu, "Measurement and modeling of video watching time in a large-scale internet video-on-demand system," *IEEE Trans. Multimedia*, vol. 15, no. 8, pp. 2087–2098, Dec. 2013.

[2] M. Cha, H. Kwak, P. Rodriguez, Y.-Y. Ahn, and S. Moon, "I tube, you tube, everybody tubes: Analyzing the world's largest user generated content video system," in *Proc. 7th ACM SIGCOMM Conf. Internet Meas.*, 2007, pp. 1–14.

[3] D. Wu, Y. Liu, and K. W. Ross, "Queuing network models for multichannel P2P live streaming systems," in *Proc. IEEE INFOCOM*, Apr. 2009, pp. 73–81.

[4] R. Zhou, S. Khemmarat, and L. Gao, "The impact of YouTube recommendation system on video views," in *Proc. 10th ACM SIGCOMM Conf. Internet Meas.*, 2010, pp. 404–410.

[5] M. Cha, H. Kwak, P. Rodriguez, Y.-Y. Ahn, and S. Moon, "Analyzing the video popularity characteristics of large-scale user generated content systems," *IEEE/ACM Trans. Netw.*, vol. 17, no. 5, pp. 1357–1370, Oct. 2009.

[6] A. Finamore, M. Mellia, M. Munafò, R. Torres, and S. Rao, "YouTube everywhere: Impact of device and infrastructure synergies on user experience," in *Proc. ACM SIGCOMM Conf. Internet Meas.*, Nov. 2011, pp. 345–360.

[7] K. Allocca, "Why videos go viral," TED Talk 2011 [Online]. Available: https://www.youtube.com/watch?v=R5BY_4FfbQs

[8] Z. Li, J. Lin, M.-I. Akodjenou, G. Xie, M. A. Kaafar, Y. Jin, and G. Peng, "Watching videos from everywhere: A study of the PPTV mobile VoD system," in *Proc. ACM Conf. Internet Meas.*, 2012, pp. 185–198.

[9] L. Chen, Y. Zhou, and D. M. Chiu, "Fake view analytics in online video services," in *Proc. ACM NOSSDAV*, 2014, pp. 1–6.

[10] A. Brodersen, S. Scellato, and M. Wattenhofer, "Youtube around the world: Geographic popularity of videos," in *Proc. 21st Int. Conf. World Wide Web*, 2012, pp. 241–250.

[11] S. Scellato, C. Mascolo, M. Musolesi, and J. Crowcroft, "Track globally, deliver locally: Improving content delivery networks by tracking geographic social cascades," in *Proc. 20th Int. Conf. World wide web*, 2011, pp. 457–466.

[12] G. Szabo and B. A. Huberman, "Predicting the popularity of online content," *Commun. ACM*, vol. 53, no. 8, pp. 80–88, 2010.

[13] J. Yang and J. Leskovec, "Patterns of temporal variation in online media," in *Proc. 4th ACM Int. Conf. Web Search Data Mining*, 2011, pp. 177–186.

[14] J. Ratkiewicz, S. Fortunato, A. Flammini, F. Menczer, and A. Vespignani, "Characterizing and modeling the dynamics of online popularity," *Phys. Rev. Lett.*, vol. 105, no. 15, p. 158701, 2010.

[15] Z. Avramova, S. Wittevrongel, H. Bruneel, and D. De Vleeschauwer, "Analysis and modeling of video popularity evolution in various online video content systems: Power-law versus exponential decay," in *Proc. 1st Int. Conf. Evolving Internet*, Aug. 2009, pp. 95–100.

[16] L. Chen, Y. Zhou, and D. M. Chiu, "Video browsing–a study of user behavior in online VoD services," in *Proc. 22nd Int. Conf. Comput. Commun. Netw.*, Jul.–Aug. 2013, pp. 1–7.

[17] H. Yu, D. Zheng, B. Y. Zhao, and W. Zheng, "Understanding user behavior in large-scale video-on-demand systems," *ACM SIGOPS Operating Syst. Rev.*, vol. 40, no. 4, pp. 333–344, 2006.

[18] F. Benevenuto, T. Rodrigues, V. Almeida, J. Almeida, and K. Ross, "Video interactions in online video social networks," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 5, no. 4, pp. 30:1–30:25, Nov. 2009.

[19] D. A. Shamma, J. Yew, L. Kennedy, and E. F. Churchill, "Viral actions: Predicting video view counts using synchronous sharing behaviors," in *Proc. ICWSM*, 2011.

[20] M. Pathan, R. Buyya, and A. Vakali, "Content delivery networks: State of the art, insights, and imperatives," in *Content Delivery Networks*. New York, NY, USA: Springer, 2008, pp. 3–32.

[21] M. Hofmann and L. R. Beaumont, *Content Networking: Architecture, Protocols, and Practice*. New York, NY, USA: Elsevier, 2005.

[22] E. J. Rosensweig, D. S. Menasche, and J. Kurose, "On the steady-state of cache networks," in *Proc. IEEE INFOCOM*, Apr. 2013, pp. 863–871.

[23] V. Martina, M. Garetto, and E. Leonardi, "A unified approach to the performance analysis of caching systems," *CoRR*, vol. abs/1307.6702, 2013 [Online]. Available: http://arxiv.org/abs/1307.6702

[24] S. Traverso, M. Ahmed, M. Garetto, P. Giaccone, E. Leonardi, and S. Niccolini, "Temporal locality in today's content caching: Why it matters and how to model it," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 43, no. 5, pp. 5–12, 2013.

Yipeng Zhou (M'05) received the B.S. degree in computer science from the University of Science and Technology of China, Hefei, China, and the M.Phil. and Ph.D. degrees in information engineering from the Chinese University of Hong Kong (CUHK), Hong Kong.

From 2012 to 2013, he was a Postdoctoral Fellow with the Institute of Network Coding, CUHK, Hong Kong. He is currently an Assistant Professor with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, China. His current research interests include modeling and analysis of large scaled networking systems, analysis of user behaviors of online video systems, and crowdsourcing-based content distribution.

Liang Chen received the double B.S. degrees from Zhejiang University, Hangzhou, China, and the Ph.D. and M.Phil. degrees in information engineering from the Chinese University of Hong Kong (CUHK), Hong Kong.

From 2013 to 2014, he was a Postdoctoral Fellow with the Department of Information Engineering, CUHK, Hong Kong. He is currently an Assistant Professor with the College of Information Engineering, Shenzhen University, Shenzhen, China. His research interests include user behavior analysis, data-driven modeling, and measurement of large-scale networking systems.

Chunfeng Yang received the B.S. degree from the Huazhong University of Science and Technology, Wuhan, China, and is currently working toward the Ph.D. degree in information engineering from the Chinese University of Hong Kong, Hong Kong.

His current research interest includes data analysis, P2P networks, online video, and video recommendation.

Dah Ming Chiu (A'86–M'03–SM'03–F'08) received the B.S. degree from Imperial College London, London, U.K., and the Ph.D. degree from Harvard University, Cambridge, MA, USA.

He is currently the Department Chairman of Information Engineering with the Chinese University of Hong Kong, Hong Kong. He previously was with Sun Labs, DEC, and AT&T Bell Labs. His current research interest includes P2P networks, network measurement, architecture and engineering, network economics, and social networks.